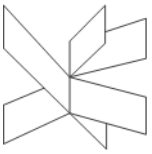


Voting System – Project Description

Eduard Fekete (355323)

Guillermo Sánchez Martínez (355442)

Alexandru Savin (354790)

Halil Ibrahim Aygun (355770)

Piotr Wiktor Junosz (355502)

Supervisors:

Ole Ildsgaard Hougaard

Michael Viuff

13942 characters

Software Technology Engineering

Second Semester

19.2.2025

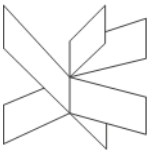
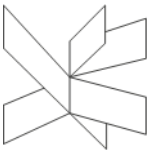


Table of contents



1. Problem Domain

Voting and Quiz Systems

Voting is a fundamental mechanism for collective decision-making and knowledge assessment in various contexts, including educational institutions, workplaces, communities, and political settings. The increasing shift toward digital solutions introduces challenges related to authentication, security, accessibility, and anonymity, necessitating a closer examination of the existing landscape and its limitations. (Gad, 2023)

Stakeholders

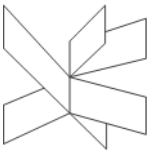
The primary stakeholders in digital voting and quiz systems include all eligible citizens of democratic countries, students, teachers, institutions, and organizations. (Anzia, 2011)

One of the important advantages of shifting elections and voting processes into digital solutions is the increase in accessibility, allowing the inclusion and targeting of larger groups of stakeholders than traditional means of voting. (Lowy, 2023)

Existing Solutions and Limitations

Several ways of digitizing these processes have been proposed and/or are currently used. All of them have certain advantages and drawbacks:

- Traditional Paper-Based Voting/Quizzing:
    - Strengths: Physical auditability, long-established trust.
    - Weaknesses: Time-consuming, expensive, and susceptible to human error. Recent events show that trust in traditional elections is on the decline in various parts of the world. (Stewart, 2022)
- Electronic Voting Machines (EVMs):
    - Strengths: Faster results, reduced human error. (McCormack, 2016)



    - Weaknesses: Security concerns, potential for tampering, limited transparency. (McCormack, 2016)
- Online Surveys and Polling Platforms (e.g., Google Forms, StrawPoll, Mentimeter, Kahoot, Quizlet):
    - Strengths: Accessibility, ease of use, fast results.
    - Weaknesses: Lack of authentication, susceptibility to multiple votes and manipulation.
- Blockchain-Based Voting:
    - Strengths: Transparency, auditability, and tamper resistance.
    - Weaknesses: Complexity, scalability issues, high computational costs.

Summary

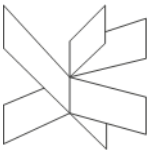
As digital participation continues to expand across multiple domains, the absence of a reliable, secure, and scalable voting infrastructure presents a significant challenge. (Gad, 2023) Current systems either compromise security, auditability, or accessibility, leading to risks such as voter fraud, data breaches, and administrative inefficiencies. A comprehensive and verifiable digital voting system is essential to mitigate these challenges while ensuring trust and integrity in democratic processes.

2. Problem statement

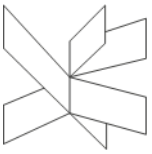
Main problem:

How might a digital voting system effectively authenticate users, prevent duplicate votes, and balance security with usability while providing a fast and seamless user experience?

Sub-questions:



- How can the system ensure that only qualified participants can access quizzes and at the same balance users' anonymity?
- How can trust in the system be established and maintained?
- How can digital voting systems handle multiple votes, bot submissions, and unauthorized access?
- What methods can be used to ensure that votes remain anonymous while still being verifiable?
- How can a voter confirm that their vote was counted without revealing their identity?
- What design principles can be used to make digital voting and quiz systems more user-friendly across different demographics?
- How can offline or low-bandwidth users participate in digital voting without compromising security?
- How can accessibility features (e.g., screen readers, voice commands, adaptive interfaces) be integrated for users with disabilities?
- How can a digital voting system handle large-scale elections with millions of participants?
- What is the impact of remote voting solutions on electoral participation?
- Open Source vs. Proprietary Debate: Should the system be open-source for transparency or closed-source for security?

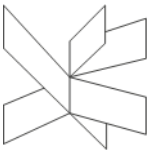


3. Delimitation

Within this project, the focus will be restricted to:

- **Software solutions**: No physical devices nor concepts of such devices will be developed to solve any part of the problem.
- **Time frame:** The solution development will only span the Semester Project subject (SEP2) length with no maintenance or updates guaranteed afterward.
- **User authentication:** Primarily a basic solution, without multi-factor authentication or advanced security protocols.

Through these delimitations, it will be ensured that the investigation maintains a clear and concise boundary, omitting all contexts that do not directly contribute to the central problem statement.



4. Choice of methods

Literature and Domain Review

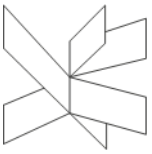
- **Method**: Targeted review of reliable articles, technical papers, and other sources of information
- **Purpose**: Gain insight into the already-built or researched solutions - their positive and adverse attributes
- **Expected Outcome**: Focus on academic databases
- **References/Tools**:
  - VIA Bibliotek
  - JSTOR
  - Nature

UML Modeling

- **Method**: Construction of UML diagrams (class, sequence, activity, etc.), capturing core domain entities and relationships between them
- **Purpose**: Creating robust object-oriented design ensuring modular code and maintainable architecture
- **Expected Outcome**: Ability to quickly capture and suggest the structure of the project
- **References/Tools**: Figma, Astah (including plugins like PlantUML), pen and paper, Literature (Larman, 2012; Gamma et al., 1994)

Design/Prototyping

- **Method**: Creating wireframes, designs, and prototypes of different fidelities from continuous user feedback, creation of personas



- **Purpose**: Achieve iterative user-centric design, incorporating rapid feedback, aligning expectations of the iteration goals.
- **Expected Outcome**: Early validation of user experience before implementation
- **References/Tools**: Figma, pen, and paper.

Iterative Development

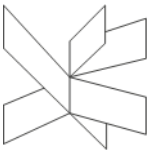
- **Method**: Manage various versions and branches of the project, ensuring clear purpose and handling of each stage of the project.
- **Purpose/Expected Outcome**: Maintain quality throughout the sprint/iteration cycles and clarity in task management.
- **References/Tools**: GitHub.

Communication, Planning, and Management

- **Method**: Centralize all necessary resources and information; ensure stable ways of communication throughout the project.
- **Purpose**: Minimize time spent on bureaucracy, provide clarity
- **Expected Outcome**: An Organized way of collaboration via self-documenting and self-sustaining platforms.
- **References/Tools**: Figma (FigJam), Discord, Messenger.

Integrated Development Environment (IDE), JavaFX/FXML Editor

- **Method**: Utilizing the interconnectedness of IntelliJ (IDE) with SceneBuilder (JavaFX/FXML Editor).
- **Purpose/Expected Outcome**: Provide a familiar and united way of implementing the solution for the development team.
- **References/Tools**: IntelliJ, SceneBuilder.



Database Management

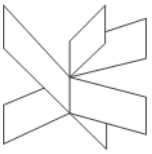
- **Method:** Utilize a dedicated database management tool for efficient data handling.
- **Purpose:** Ensure structured and optimized database interactions throughout development.
- **Expected Outcome:** Well-organized and maintainable database structure supporting the project requirements.
- **References/Tools:** DataGrip, PostgreSQL.

Apply Scrum in combination with Unified Process (UP)

- **Method:** Apply the iterative Scrum process together with the disciplined phases of the Unified Process to organize the development life cycle.
- **Purpose:** Combine flexible, sprint-based progress monitoring with clearly established system phases to raise the quality and controllability of the project.
- **Expected Outcome:** Continuous enhancement with clear milestones and set requirements throughout the project, along with reacting to change and delivering in incremental increments.
- **References/Tools:** Literature(Schwaber & Sutherland, 2020).

Testing

- **Method:** Implement testing to ensure correctness and reliability.
- **Purpose:** Detect and resolve errors early in the development cycle.
- **Expected Outcome:** High confidence in the functionality of each part of the system, leading to fewer bugs in later stages and easier maintenance.
- **References/Tools:** ZOMBIES testing (Maximiliano Contieri, 2020)



5. Time schedule

Final Deadline

- **Date:** May 28, 2025

Events

1. Sprint Planning & Task Assignment

- **When**: At the start of every sprint
- **Details**: Set sprint goals, assign important backlog items for each iteration

2. Daily Scrum

- **When**: In the beginning of every day, during the sprint.
- **Platform**: Conducted through Discord or in person at school.
- **Purpose**: Quick 15-minute talk to discuss progress, blockers, and next steps.

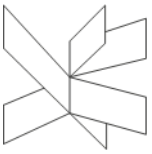
3. Sprint review and retrospective

- **When**: At the end of each sprint.
- **Details**:
    - **Sprint Review**: Show completed work and ensure the product is on the right track (product-oriented).
    - **Retrospective**: Reflect on team collaboration, identify issues, and make suggestions on how to improve efficiency and workflow for future sprints (process-oriented).

3. Completion of Formal Project Part

- **Target Date:** May 20, 2025

- **Details:** Aim to complete the formal writing and documentation aspect by this date, allowing time for final revisions before the deadline.



Expected Time Commitment Based on 10 ECTS

Each student is expected to contribute a total of 275 hours to meet the 10 ECTS workload requirement, with increased hours in March and April to minimize the workload in May.

Breakdown of Hours for 10 ECTS:

- **March:**

- Weekly commitment: 20 hours per student

- **Total for March:** 4×20=80 hours per student

- **April:**

- Weekly commitment: 20 hours per student

- **Total for April:** 4×20=80 hours per student

- **May (up to May 20):**

- Remaining:115 hours

- Weekly commitment: 115/3 ≈ 38 hours per student

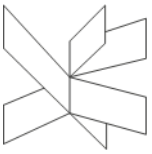
- **Total for May:** 3×38 =114 hours per student

Total Hours Calculation for 10 ECTS

- **Total Hours:** 275 hours per student

- **Calculation:**

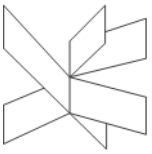
- Total hours = 10×27.5=275 hours



- This distribution ensures the project meets the workload requirement for 10 ECTS, with a reduced workload in May due to a higher time commitment in March and April.

Summary

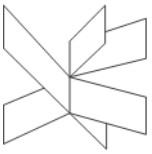
For each student, it amounts to 275 hours of work to achieve 10 ECTS.



6. Risk assessment

Risk	Likelihood	Severity	Normalized product	Preventive Actions	Identifiers	Responsible
Scope creep	3	2	1.7	Frequent meet-ups and effective communication	Misunderstanding in task “completeness”, Misalignment and disagreement during meetings	Ibrahim
Communication Issues	3	3	2.3	Agreeing on communication media, communicating high-importance details physically or during calls	Long waiting periods for replies, editing/working over each other’s work	Alexandru
Lack of knowledge	2	2	1.5	Keeping up with other subjects, studying frequently, and asking supervisors.	Lack of understanding and contribution to the project. Failing the other SWE subjects.	Eduard
Poor sprint planning	3	3	2.3	Using a well organized Product Backlog and involve the whole team in planning	Unclear scope, wasted effort and time mismanagement	Piotr
Lack of documentation	2	2	1.5	Maintaining consistent documentation	Slower progress of development caused by the lack of consistency	Guillermo

Likelihood, Severity, and Normalized product are on a scale of 1-5 with the normalized product being the multiple of likelihood and severity normalized to fit the scale.



7. References

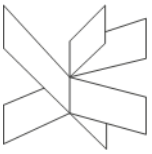
Anzia, S. F. (2011). Election Timing and the Electoral Influence of Interest Groups. *The Journal of Politics*, *73*(2), 412–427. https://doi.org/10.1017/s0022381611000028

Gad, C. (2023). Trusting elections: complexities and risks of digital voting in Denmark. *Journal of Cultural Economy*, 1–18. https://doi.org/10.1080/17530350.2023.2246991

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design patterns : elements of reusable object-oriented software*. Addison-Wesley.

Larman, C. (2012). *Applying UML and Patterns: An Introduction to Object Oriented Analysis and Design and Interative Development: 3rd Edition*. Pearson Education India.

Lee, K., James, J. I., Ejeta, Tekachew Gobena, & Kim, H. J. (n.d.). ELECTRONIC VOTING SERVICE USING BLOCK-CHAIN. *The Journal of Digital Forensics, Security and Law : JDFSL*, *11*(2), 123–135. https://www.proquest.com/docview/1825910293?pq-origsite=primo&sourcetype=Scholarly%20Journals



Lowy. (2023). Elections. In *Building the AustraliaPNG Digital Ecosystem* (pp. 15–15). Lowy Institute for International Policy; JSTOR. https://doi.org/10.2307/resrep51373.12

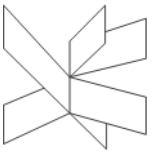
Maximiliano Contieri. (2020, November 11). *Zombie Testing: One Behavior at a Time*. Hackernoon.com. https://hackernoon.com/zombie-testing-one-behavior-at-a-time-9s2m3zjo

McCormack, C. B. (2016). Electronic Voting around the World. In *Democracy Rebooted* (pp. 9–14). Atlantic Council; JSTOR. https://doi.org/10.2307/resrep03645.9

Panchbudhe, S., Shaikh, S., Swami, H., Kadam, C. Y., Padalkar, R., Shivkar, R. R., Gulavani, G., Gulajkar, S., Gawade, S., & Mujawar, F. (2024). Efficacy of Google Form–based MCQ tests for formative assessment in medical biochemistry education. *Journal of Education and Health Promotion*, *13*(1), 92. https://doi.org/10.4103/jehp.jehp_981_23

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide The Definitive Guide to Scrum: The Rules of the Game*. https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf

Stewart, C. (2022). Trust in Elections. *Daedalus*, *151*(4), 234–253. JSTOR. https://doi.org/10.2307/48696783



Wang, A. I., & Tahir, R. (2020). The Effect of Using kahoot! for Learning – a Literature Review. *Computers & Education*, *149*(149), 103818. https://doi.org/10.1016/j.compedu.2020.103818